REMARKS

This amendment is responsive to the Office Action dated February 3, 2006. Applicants have amended claims 24, 45, 47 and 53 to correct certain informalities. Applicants have also cancelled claims 9, 10 and 36. Claims 1-8, 11-35 and 37-53 are pending.

Claim Rejection Under 35 U.S.C. § 102

In the Office Action, the Examiner formed new grounds for the rejection. In particular, the Examiner rejected claims 1-8, 11, 45-47, 51 and 53 under 35 U.S.C. 102(e) as being anticipated by Ellis (USPN 6,484,257). Applicants respectfully traverse the rejection. Ellis fails to disclose each and every feature of the claimed invention, as required by 35 U.S.C. 102(e), and provides no teaching that would have suggested the desirability of modification to include such features.

For example, Applicants' claim 1 requires managing a communications negotiation between the client and the server through an intermediate device that supports both a direct mode and a proxy mode. Further, claim 1 requires decrypting encrypted data packets, and forwarding unencrypted data packets from the intermediate device to the server using a communication session *negotiated by the client and the server* when the intermediate device operates in direct mode. Claim 1 further requires forwarding unencrypted data packets from the intermediate device to the server *using a second communication session negotiated by the server and the intermediate device* when the intermediate device operates in proxy mode. That is, depending on the mode, the Applicants' claimed intermediate device either transparently uses the same communication session negotiated by a client and a server to forward decrypted data to the server (direct mode), or uses a separate session negotiated by the intermediate device and the server (proxy mode).

In rejecting claim 1, the Examiner asserts that Ellis describes an intermediate device that supports both a direct mode and a proxy mode as defined by Applicants' claims.[1] As a basis for this assertion, the Examiner cites Ellis at col. 7, ln. 11 – col. 8. ln. 27 without further explanation. However, as explained by Applicant in further detail below, none of the components of the Ellis

[1] Office Action, pg. 2.

system operates as an intermediate device that supports both a direct mode and a proxy mode as defined by Applicants' claims.

In the passage cited by the Examiner, Ellis describes a distributed computing environment having a main server, agent servers and clients. Both the main server and the agent servers are enabled to receive secure connections.[2] Ellis describes two general scenarios: (1) handling new connections, and (2) redirecting existing connections.

With respect to a client request for a new connection, Ellis states that the main server first determines whether it can handle the new connection (session).[3] If so, the main server begins the session.[4] If the main server cannot handle the new connection, then the main server "wakes up" an agent server to handle the requested connection. In the event the client specified a particular one of the agent servers (referred to as a "peer agent server") to connect to directly, then the main server attempts to redirect the client to the peer agent server after authenticating both the client and the agent server. In either case, the agent server determines whether it can handle the requested connection. If so, the agent server and the client then negotiate the new session.[5] The agent server then receives encrypted session communications from the client via the negotiated session, decrypts the session communication and forwards the decrypted data to its proper destination.

It is reasonably clear from the Ellis description that, with respect to new sessions, the client and the agent server are responsible for negotiating the new session. Ellis makes reasonably clear that the agent server, as an intermediate device, negotiates the new connection (session) with the client, decrypts data received from that connection and forwards the decrypted data. Thus, with respect to new connections, the main server and the agent servers in Ellis appear to be operating as the classical proxy servers on behalf of destinations, and certainly do not use sessions negotiated by the client and the destination.

[2] Ellis at col. 7, ln. 22.
[3] Ellis at col. 7, ln. 24.
[4] Ellis at col. 7, ln. 29.
[5] Ellis makes it abundantly clear that the client and the agent server negotiate the session. Ellis at col. 7, ln 54, for example, states that the client and the agent "independently generate" session keys for new session. At col. 8, ln. 46, Ellis describes 6 steps of an "overall system algorithm." In Step 5, Ellis states that the client and the agent server "negotiate" the session key and the proper security association. Ellis then illustrates each packet having an AGENT IP HEADER 5A24, which is a clear indication that the session is between the client and the Agent.

With respect to the second scenario, i.e., redirecting existing connections, Ellis states that the main server may "pass" an existing session from one agent server to another different agent server.[6] If an agent server becomes saturated, for example, it notifies the main server to "pass the session on to another agent server."[7] During this process, the main server notifies the client, and the client connects to the new agent server.[8] The new agent server uses security information for the session to continue the secure session with the client. That is, the new agent server has replaced the role of the original agent server. The previous agent server then closes the original session.[9]

In this regard, it is reasonably clear that the new agent server replaces the role of the previous agent server as operating as an end-point for a session with the client. Therefore, even for redirected existing sessions, the new agent server again operates as a proxy server by receiving encrypted communications from the client via a session negotiated by the agent server and that client, decrypting the communications, and then forwarding the decrypted communications to the ultimate destination.

For at least these reasons, Ellis fails to teach or suggest an intermediate device that supports both a direct mode and a proxy mode. In particular, claim 1 requires decrypting encrypted data packets, and forwarding unencrypted data packets from the intermediate device to the server using a communication session *negotiated by the client and the server* when the intermediate device operates in direct mode. In contrast, all references in Ellis to negotiation of a session refer to the client and the intermediate device (agent server). For example, Ellis at col. 7, ln. 54 states the client and the agent "independently generate" session keys. At col. 8, ln. 46, Ellis describes six steps of the "overall system algorithm." In Step 5, Ellis states that the client and the agent server "negotiate" the session key and the proper security association. Ellis then illustrates each packet output from the client as having an AGENT IP HEADER 5A24, which is stripped off and entirely replaced by a different DESTINATION IP HEADER 5B30. This is a clear indication that, in Ellis, the session used by the agent server to forward data to the destination was not negotiated by the client and the destination.

[6] Ellis at col. 8, ln. 8.
[7] Ellis at col. 8, ll. 8-9.
[8] Ellis at col. 8, ll. 14-15.
[9] Ellis at col. 8, ll. 21-22.

There is no teaching or suggestion in Ellis of an intermediate device located between a client and a server, where the intermediate device operates in a direct mode to decrypt encrypted data packets and forward unencrypted data packets from the intermediate device to the server using a communication session *negotiated by the client and the server*. In contrast, the intermediate devices (agent servers) of Ellis only operate as proxies that separately negotiate communication sessions with clients on behalf of destinations.

For at least these reasons, Ellis fails to anticipate the requirements of independent claims 1, 33 and 45. Moreover, none of the other references, either singularly or in combination, provide any teaching or suggestion that overcomes the deficiencies of Ellis.

With respect to dependent claim 2, Ellis provides no teaching of modifying negotiation data received from a client prior to forwarding the negotiation data to the destination. In rejecting claim 2, the Examiner cited Ellis at col. 8, ln. 28-53. As described above, this passage describes six steps of the "overall system algorithm." In Step 5, Ellis states that the client and the agent server "negotiate" the session key and the proper security association. There is no indication in Ellis whatsoever that either the main server or the agent server modifies negotiation data and then forwards the modified negotiation data to the destination. To the contrary, the agent server negotiates the session directly with the client, as clearly stated in step 5 of the Ellis algorithm.

With respect to dependent claim 3, nothing in Ellis suggests modifying a SYN request. In fact, in its entirety, Ellis does not even refer to a SYN request, let alone modification of a SYN request by an intermediate device.

With respect to dependent claims 51 and 53, the Examiner asserted that Ellis teaches *automatically switching* the intermediate device from the direct mode to the proxy mode upon detecting a communication error associated with the direct mode. In rejecting claims 51 and 53, the Examiner again cited Ellis at col. 7, ln. 11 to col. 8, ln. 27 without comment.

As discussed above, the portion of Ellis describes agent servers that solely operate as proxies and does not teach or suggest an intermediate device that supports a direct acceleration mode at all. Moreover, the Applicants are at a loss as to where the Examiner finds a teaching for an intermediate device that *automatically switches* from the direct acceleration mode to the proxy mode upon detecting a *communication error* associated with the direct mode. The portion of

Ellis cited by the Examiner does not even mention communications errors or switching between modes. Ellis only states that if an agent server becomes saturated, then an existing session may be moved <u>to a different agent server</u>. This does not anticipate an intermediate device that automatically switches acceleration modes for multiple reasons. First, as discussed above, the agent servers of Ellis operate only as proxies and do not even support a direct mode. Second, Ellis requires moving an existing session from one agent server to another. This is entirely different from changing an acceleration mode of a given intermediate device handling the session.

Ellis fails to disclose each and every limitation set forth in claims 1-8, 11, 45-47, 51 and 53. For at least these reasons, the Examiner has failed to establish a prima facie case for anticipation of Applicants' claims 1-8, 11, 45-47, 51 and 53 under 35 U.S.C. 102(e). Withdrawal of this rejection is requested.

<u>Claim Rejection Under 35 U.S.C. § 103</u>

In the Final Office Action, the Examiner rejected claims 9, 10, 12-44, 48-50 and 52 under 35 U.S.C. 103(a) as being unpatentable over Ellis in various combinations with other references. Applicants respectfully traverse the rejections. The applied references fail to disclose or suggest the inventions defined by Applicants' claims, and provide no teaching that would have suggested the desirability of modification to arrive at the claimed invention.

Independent claim 20

In the Office Action, the Examiner rejected claims 20-22, 27 and 29 under 35 U.S.C. 103(a) as being unpatentable over Ellis (USPN 6,484,257) in view of Maloney et al. (USPN 6,253,337). Applicants' independent claim 20 recites steps performed by an intermediary device when using the "direct" (cut through) communication session described above. For example, claim 20 requires establishing a communications session between the client and said one of said plurality of servers by receiving negotiation data from the client intended for the server and forwarding the negotiation data in modified form to the server, and receiving negotiation data from the server intended for the client and forwarding the negotiation data to the client <u>to establish the client and the server as terminations for the communications session</u>.

In this manner, claim 20 requires that a communication session is established through the intermediary device by the client and the server, and the client and the server operate as end-points for the TCP/IP session. In addition, claim 20 requires establishing a secure communications session between the client and the intermediary device, and forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server.

For reasons set forth above, Ellis fails to describe an intermediate device (an agent server in Ellis) that forwards decrypted application data to said one of said plurality of servers *using the communications session established between the client and the server*, as required by claim 20. Ellis provides no teaching whatsoever of a session negotiated by the client and the ultimate destination where those devices operate as end-points for the session, and certainly fails to describe an intermediate device that uses that same session to forward decrypted data to the server. As discussed in detail above, the intermediate devices (agent servers) of Ellis only operate as proxies that separately negotiate communication sessions with clients on behalf of the destinations.

In rejecting claim 20, the Examiner cited Ellis at col. 8, ln. 54 – col. 9, ln. 49 without further comment. This passage of Ellis describes FIGS. 5A, 5B of Ellis, which show the network-layer processing of packets. As discussed, the previous passages of Ellis make clear that the Server Agents are responsible for negotiating sessions with the clients, and this portion of Ellis is entirely consistent with this fact. In particular, this portion of Ellis describes the Server Agent as "stripping" incoming IP information (the AGENT IP HEADER, ESP, AH and AGENT ID IP HEADER layers) and pre-appending entirely new IP information (DESTINATION IP HEADER 5B30) for the final destination host 5B40.

Thus, it is abundantly clear that that client in Ellis is not communicating using a session negotiated by the client and the destination host. Otherwise, the client would not output packets to have AGENT IP HEADER and AGENT ID IP HEADER layers, and these layers would not have to be entirely replaced with different IP information. The client would, in fact, have little or no knowledge that the Server Agent were operating as an intermediate device. The clients in Ellis are, in fact, communicating with the agent servers as proxies using sessions negotiated with those agent servers, and those agent servers decrypt data and forward the clear text to the destination using entirely different IP header information. For at least this reason, Ellis in view of the other references fail to teach or suggest forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session *established between the client and the server.*

Applicant again refers the Examiner to the present application that describes the direct mode, also referred to as a "direct" version of a "cut through mode," as follows:

> Figure 5 illustrates a direct, cut through processing method. Packets from client to server are addressed from the client to the server and from server to client, with the intermediary, SSL device being transparent to both. In the embodiment shown therein, the SSL accelerator allows the client and server to negotiate the TCP/IP session directly, making *only minor changes to the TCP/IP headers* passing through the accelerator device, and tracking session data in a data structure in memory to enable SSL session handling to occur. *As described herein, this mode is referred to herein as the "direct, cut-through" mode, since the client and server "think" they are communicating directly with each other, and the SSL accelerator is essentially transparent.*

As illustrated in FIG. 5, client 100 and server 300 negotiate the TCP/IP session and operate as termination points for the session. In other words, in "direct, cut through" (direct) mode, the intermediate acceleration device still handles functions required for the SSL session, while the client and the server handle the TCP/IP session. Since the client and server handle the TCP/IP session, unlike Ellis, only minor changes are made by Applicants' intermediate device in direct mode. Much of the session information, such as the sequence numbers, need not be changed by Applicants' intermediate device in direct mode, thereby preserving processing power and limiting delay. In contrast, Ellis utilizes agent servers as proxies that replace the entire Agent IP header with Destination IP header.

Independent claim 33

In rejecting independent claim 33, the Examiner again primarily relied on Ellis. Applicants' independent claim 33 requires an acceleration apparatus adapted to operate in either one of a direct mode and a proxy mode. Claim 33 requires that in the direct mode the intermediate *acceleration apparatus decrypts data packets received from the client and forwards the decrypted data packets to one of the servers using a communication session negotiated by the client and the server*, and in the proxy mode the acceleration apparatus responds to the client on behalf of the server and forwards the decrypted data packets to the server using a communication session negotiated by the acceleration device and the server. For reasons set forth above, Ellis does not describe an acceleration apparatus that support a proxy mode in combination with a direct mode as claimed by the Applicants.

With respect to Applicants' dependent claims, none of the other references, either singularly or in combination, address this basic deficiency of the prior art with respect to an intermediate acceleration device that combines a full proxy mode with a direct mode. None of the references, either singularly or in combination, teach or suggest an intermediate device in which the intermediate device transparently provides encryption and decryption services, but uses the same communication session that the client and server originally negotiated to forward decrypted data to the server. For example, Cohen describes a proxy architecture that uses separate TCP connections and fails to describe a direct or cut through mode. Maloney et al. describes an information analysis system that is a combination of sensor, analysis, data conversion, and visualization programs. Boeuf describes a file server that stores data by allocating a single oversized contiguous storage area and by allowing data wrapping. Fujiyama et al. describes a network system in which each of multiple networks, each containing computers and relay computers, is connected to another network via multiple relay computers. None of the relay computers act as acceleration devices. Bellaton et al. describes a mechanism for dispatching a sequence of packets via a telecommunications network. Gelman et al. describes a method of communicating over a satellite or other high delay-bandwidth link that does not utilize TCP/IP. Holtey et al. describes a secure memory card and is unrelated to a network acceleration device. Harper et al. describes techniques for rejuvenating a component of a distributed data processing environment.

Application Number 09/900,515
Responsive to Office Action mailed March 3, 2006

CONCLUSION

All claims in this application are in condition for allowance. Applicants respectfully request reconsideration and prompt allowance of all pending claims. Please charge any additional fees or credit any overpayment to deposit account number 50-1778. The Examiner is invited to telephone the below-signed attorney to discuss this application.

Date:

May 22, 2006
SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

By:

Name: Kent J. Sieffert
Reg. No.: 41,312